

**GTD aplikace s možností
synchronizace na mobilní telefon**
**GTD Application with Mobile Phone
Synchronization Features**

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

V Ostravě 7. května 2010

.....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2010

.....

Rád bych na tomto místě poděkoval všem, kteří mi s prací pomohli, protože bez nich by tato práce nevznikla.

Abstrakt

Organizace práce je dnes záležitostí, nad kterou by se měl každý alespoň na chvíli zamyslet. Tato bakalářská práce pojednává o tom, proč je upořádání času tak závažnou problematikou a nabízí lehký úvod do řešení, jak se s těmito problémy vyrovnat. Na začátku práce jsou popsány základní principy a osvědčené praktiky, které pomáhají bojovat proti stresu. Dále je rozebrána podrobněji jedna z nejznámějších metod této problematiky - Getting Things Done. Cílem práce je navrhnout a implementovat aplikaci, která bude principy této metody využívat a uvede je do praxe. Zároveň bude tato aplikace poskytovat možnost synchronizace s již existující mobilní aplikací, využívající tuto metodu. Další kapitoly tedy popisují návrh a implementaci aplikace. Závěr je věnován implementaci synchronizace na obou zařízeních.

Klíčová slova: Time management, GTD, Aplikace, GUI, Python, PyQt, Android, Synchronizace, Git

Abstract

Today's time management is not an easy task. This bachelor thesis explains, why this is such a big thing and tries to provide some kind of a conclusion about how to deal with this problems. From beginning, it describes basic principles and practics, which can help to stay in sanity while in unceasing fight against stress. These practics are well-established by the people who were facing the same problem already before. Following chapter analyses one of the best method afield of the time management called Getting Things Done. The main goal of this thesis is a project consisting of designing and implementing an application based on GTD and synchronization with already existing Android appliacion. Next chapters are about how to design and implement this goal and in the end of this thesis is description of the implementation of synchornization.

Keywords: Time management, GTD, Application, GUI, Python, PyQt, Android, Synchronization, Git

Seznam použitých zkratk a symbolů

GTD	– Getting things done
GUI	– Graphical user interface
Qt	– Multiplatformní knihovna pro vyvíjení grafického uživatelského rozhraní
PyQt	– Qt knihovna pro skriptovací jazyk Python
DFD	– Data flow diagram
SŘBD	– Systém řízení báze dat
OSS	– Open-source software
SDK	– Software development kit
ADT	– Android development tools
IDE	– Integrated development environment
URL	– Uniform resource locator
HTTP	– Hypertext transfer protocol
XML	– Extensible markup language

Obsah

1	Úvod	5
1.1	Motivace	5
2	Metody organizace práce	6
2.1	Generace time managementu	6
2.2	Základní techniky	6
2.3	Filozofie	7
3	Getting things done	10
3.1	Potřeba vyšší osobní produktivity	10
3.2	Základní přínos a cíl metody	10
3.3	Autor metody – David Allen	11
3.4	Pět kroků k efektivnímu využití svého pracovního času	11
3.5	Zhodnocení metody GTD	19
4	Nárvh	20
4.1	Návrh desktopové aplikace	20
4.2	Návrh synchronizace	24
5	Implementace	27
5.1	Implementace desktopové aplikace	27
5.2	Implementace synchronizace na desktopové aplikaci	29
5.3	Implementace synchronizace na mobilní aplikaci	32
6	Závěr	34
7	Reference	35

Seznam tabulek

1	Synchronizace bez konfliktu	25
2	Konfliktní synchronizace	26

Seznam obrázků

1	Stromový model druhého a třetího kroku GTD metody [1, str. 32]	14
2	Ukázka mobilní aplikace	21
3	Konceptuální model desktopové aplikace	22
4	Smazání projektu	24
5	Vyhledání nejbližších úkolů projektu	24
6	Ukázka desktopové aplikace	28
7	Ukázka konfliktu na desktopové aplikaci	31
8	Ukázka synchronizace na mobilní aplikaci	33

Seznam výpisů zdrojového kódu

1	Vytvoření nového repositáře	29
2	Vytvoření vzdáleného repositáře	30
3	Pokus o synchronizaci se vzdáleným repositářem	30
4	Zjištění obsahu vzdálené databáze	30
5	Vyřešení konfliktu	30
6	Nahrání změn do nové větve vzdáleného repositáře	32
7	Pokus o sloučení hlavní větve s vedlejší	33
8	Zjištění rozdílů mezi databázemi	33

1 Úvod

Organizace práce je dnes záležitostí, nad kterou by se měl každý alespoň na chvíli zamyslet. Tato problematika je však natolik obsáhlá, že by tato bakalářská práce nestačila ani na její část. Proto lze tuto práci brát pouze jako jednoduché natuknutí problému a jeho částečné vysvětlení. Jsou zde pouze jednoduše shrnuty nejdůležitější a nejznámější praktiky na ovládnutí času. Už vůbec nelze brát tuto práci jako návod, jak tyto problémy vyřešit nebo jak být více produktivní nebo efektivní. Řešením stresu a zvládáním času se zabývají mnohastránkové knihy [1]. Práci lze vnímat jako jakýsi obsah, nastínění a náhled, jak by takové zvládání mohlo probíhat.

Teoretická část je tedy věnována organizaci času a produktivitě. Jsou zde rozebrány jednotlivé metody a přístupy k těmto problematikám. Tato práce nevykládá uvedené principy za správné, pouze konstatuje fakta a zjištěné informace. Dále je v textu popisován složitější model pro organizaci práce zvaný Getting Things Done.

Praktická část se skládá z návrhu a implementace aplikace, která má pomáhat ve snaze zorganizovat si vlastní čas. Na tomto poli již existuje několik aplikací, proto je navržena tak, aby se dokázala synchronizovat s jednou z těchto už existujících aplikací. Návrh a implementace aplikace je zde podrobně rozebrán.

1.1 Motivace

S příchodem nových technologií, jako je televize, telefon či internet se tok informací proudících do naší hlavy za několik let rapidně zvýšil. Tyto výdobytky doby s sebou nepřinášejí zpravidla jenom pozitivní věci. Jednou z těch negativních věcí je mimo jiné také stres, který je v dnešním hektickém a uspěchaném světě součástí každodenního života. Skoro každý jedinec je denně vystavován stresujícím situacím, tlakům a zkouškám, ať už v důsledku pracovního nebo jiného vypětí. Kdo by dnes nechtěl hodit všechny trable za hlavu, nedělat si z ničeho starosti, nestresovat se. Jistě existuje několik lidí, kteří se k takovému řešení uchýlili, ale tito lidé pouze utíkají před reálným světem a často se stávají terčem posměchu druhých. Sám si nemyslím, že je na tom něco špatného, je však lepší se k problémům stavět čelem. Člověk není biologicky vybaven, aby dokázal dnešnímu stresu čelit. Aby přežil, přizpůsobil se, musí se s neustálými změnami vyrovnat po svém a vyvíjet se s dobou.

Stres se stal pro většinu lidí nepřítelem číslo jedna. Mnoho z nás nestíhá plnit své úkoly, dnes řeší co měli udělat včera, v horším případě před týdnem nebo před delší dobou. Mnoho lidí se snaží provést více práce než dokáže zvládnout nebo problémy odkládá. Mnozí mají také problémy se spaním, jelikož před usnutím probírají a vzpomínají na věci, které nestihli nebo které je teprve čekají. Takoví lidé potom ani nemají většinou čas na odpočinek a relaxaci, nedají si oddych a nedopřejí si klid. Tento životní styl vede v lepším případě k totálnímu vyčerpání a nemocem, v tom horším třeba i do psychiatrické léčebny.

Člověk nedokáže a nemůže stihnout všechno. Je třeba si říct ne, některé věci přesunout, jiné úplně zrušit, určit si priority. Naštěstí se několik chytrých lidí už těmito problémy zabývalo před námi. Tito lidé poskytují návod jak se s takovými záležitostmi vyrovnat, jak je řešit. Tato problematika se obecně nazývá time management neboli organizace času.

2 Metody organizace práce

Time management je soubor znalostí, nástrojů a technik, který definují jak správně nakládat s časem a zvládnout všechny úkoly, projekty a cíle. Zahrnuje širokou škálu činností jako je plánování, rozvrhnutí, určení cílů, delegování, analýza stráveného času, sebekontrola, organizování, postup a určení priorit. Na počátku se time management používal pouze pro obchodní či pracovní činnosti, ale postupem času rozšířil pole své působnosti i mezi záležitosti osobní [5]. Je důležité si uvědomit, že time management není o tom, jak stihnout spoustu věcí. Mnohem důležitější je jistota, že pracujeme na těch správných věcech, které potřebují být doopravdy realizovány.

2.1 Generace time managementu

Existují stovky metod a přístupů k time managementu. Rozdělujeme je do čtyř generací [5].

První generace používá jakýsi *to-do-list* pro zapisování všech činností a úkolů, které je potřeba udělat. Činnostem nepřirazuje žádnou prioritu. Mezi hlavním cílem a položkami neexistuje žádný vztah což způsobuje tendenci úkoly vykonávat bezmyšlenkovitě.

Druhá generace se snaží více plánovat dopředu a rozvrhnout úkoly a nadcházející události v kalendáři. Klíčovým prvkem je pracovní plán na celý den.

Třetí generace upřesňuje hodnoty, zavádí priority a cíle. V závislosti na tom již lze od sebe systematicky rozlišovat urgentní a důležité položky v seznamu. Položky v diáři nyní mají počáteční a koncové datum.

Čtvrtá generace se soustředí více na způsob, jakým dosáhnout cíle. Zabývá se raději těmi důležitějšími činnostmi a řeší problémy spíše obecněji a v dlouhodobém měřítku než jednotlivě. Také je důležité, aby se člověk cítil dobře a spokojeně, pozitivně smýšlel a na základě toho se stává více produktivním.

2.2 Základní techniky

2.2.1 To-do list

To-do list neboli seznam úkolů je soupis činností, které je třeba provést [5]. Může se jednat jak o sadu kroků pro velký projekt, tak i o seznam domácích prací. Jedna listina může obsahovat i více seznamů, potom se jedná o projekty, a tyto seznamy mohou obsahovat další seznamy a postupně se do sebe zanořovat až k jednotlivým úkolům. Pokud máme úkol splněn, jednoduše se škrtně či vymaže. Hlavní funkcí takového seznamu je sloužit jako náhrada naší paměti. Většinou to může být jednoduchý list papíru, bloček nebo diář. Existují různé alternativy i pro mobilní telefony a osobní počítače. To-do listem může být i kalendář, kde zapisujeme naplánované události.

2.2.2 ABC priority

Tato technika byla po dlouhou dobu používána pro řízení větších firem jako kategorizace velkých objemů dat do skupin. Tyto skupiny jsou často označovány písmeny A, B a C [5].

Pro větší rozvrstvení dat může být skupin i více. Základem techniky ABC je to-do list se všemi úkoly. Každá činnost se na základě priority přidělí do jedné ze skupin.

- A - Úkoly, které se za každou cenu musí dnes dokončit. Pokud je úkol příliš složitý na to, aby ho bylo možné dokončit za den, rozdělí se na menší části, které se znovu podrobí analýze. Pokud činnost není třeba dnes dokončit, přesune se do kategorie 'B'.
- B - Zde je místo pro důležité úkoly, události nebo termíny, které vyžadují nějakou činnost, ale nemusí být dokončeny dnes. Například koupit lístky na zápas, který se odehraje za týden.
- C - Do této skupiny patří všechny činnosti, které nemusí být dokončeny v nějakém čase. Tento seznam se využívá většinou pro úkoly, které jsou vykonávány s potěšením, pro obluby a zájmy. Pokud předchozí dvě skupiny neobsahují žádné úkoly, nebo z nějakého důvodu tyto úkoly nelze zrovna vykonávat, lze pracovat na záležitostech z této skupiny.

2.3 Filozofie

Čas nelze koupit. Času máme každý stejně a je pouze na nás, jak ho dokážeme využít. Proto je třeba se nad touto problematikou zamyslet a čas si rozvrhnout pokud možno co nejefektivněji. Lidé většinou dopředu nepřemýšlejí a řeší problémy s tím, jak přicházejí. Až v časové tísní uvažují, kde udělali chybu, což je stojí o mnoho více energie a stresu než při průběžném plánování.

Další informace o problémech organizace práce lze nalézt na internetu [3, 4].

2.3.1 „Čeho chci dosáhnout?“

Je nezbytně nutné určit si osobní cíle a priority a nedopustit, aby byly vytěsněny urgentními problémy [2]. Tyto cíle je potřeba si někde zaznamenat a každý den si je připomínat, například lepícím štítkem na obrazovce počítače nebo programem v počítači, či na ledničce na kusu papíru pomocí magnetu. Nezáleží na způsobu, ale na konečném výsledku. Někteří lidé používají jakýsi druh mantry, při kterém si před spaním a po probuzení opakují, čeho by chtěli dosáhnout. Pozor, problémy ovšem nerozebírají dopodrobna, pouze si představují výsledek, ke kterému by chtěli dospět.

Podle doby potřebné k provedení daného úkolu se činnosti rozlišují na různé tzv. letové hladiny. Koupit deset rohlíků a půl kila brambor bude určitě úkol s velmi nízkou hladinou, naproti tomu předsevzetí stát se prezidentem bude hodně vysoko. V závislosti na letové hladině se problémy mohou postupně dělit na projekty, plány nebo jednoduché seznamy úkolů. Jednotlivé úkoly a cíle mohou mít různé úrovně důležitosti, různé priority nebo termíny, do kdy je třeba danou činnost dokončit.

2.3.2 „Chci stihnout vše“

Asi nejdůležitější záležitostí v souvislosti s časovým omezením je umět říci „ne“ méně důležitým věcem, neprioritním projektům. Nebudíme otroky své práce. Můžeme pracovat 24 hodin denně, ale stejně vše co chceme, nedokážeme stihnout. A ne jenom, že vše nestihneme. Větším upnutím k práci začneme zanedbávat jiné základní činnosti a dostaneme se více a více do stresujících situací. Člověk není stroj a málokdo dokáže soustředěně pracovat několik hodin denně. Jelikož jsou ale cíle již určeny, nebude problém přebytečné aktivity vyškrtnout a věnovat svůj čas pouze těm prioritním.

2.3.3 „Proč to dělám?“

Další důležitou věcí je vědět, proč vůbec něco dělat. Co mi daná akce přinese a jak se na to budu dívat za týden, měsíc nebo rok? Má cenu dělat zbytečné, neužitečné činnosti? Určitě ne, ale ne každý si toto uvědomuje. Pokud neznáme odpověď na otázku proč, nemá smysl se danou aktivitou vůbec zabývat. Jedno z pravidel, které se v time managementu používá, říká, že 80% vykonané práce přinese 20% zisku. Zbylých 20% naopak vynesou 80% zisku, ať už se jedná o práci, studium, relax či mezilidské vztahy [5]. Neexistuje ovšem žádné obecné pravidlo, které by odhalovalo, které činnosti patří do té lepší či horší části. Nejjednodušším pravidlem je řídit se subjektivním dojmem a vycházet ze stanovených cílů a priorit.

2.3.4 „Stihnu více věcí najednou“

Koncentrace je při řešení důležitých úkolů jedna z klíčových věcí. Práce bez koncentrace znamená většinou několik hodin času navíc. Proto není vůbec dobré myšlenky odvádět a řešit jiné věci, které se netýkají současné činnosti nebo ještě hůře, řešit několik věcí najednou. Tzv. multitasking většinou znamená, že zvládneme udělat kousek od každého, ale celkově vůbec nic, což je velice neefektivní. Nikdy se nedostaneme do hloubky jednoho problému, protože mezitím přeskochíme k problému dalšímu. Nebo naopak děláme jednu z činností bezmyšlenkovitě. To samé platí i v případě používání dnešních vymožeností jako jsou telefony, televize a internet. Pokud budu při práci každých pět minut zjišťovat, co nového se děje na internetu, sledovat oblíbený seriál, nebo mi co chvíli zazvoní telefon, nikdy nebudu na 100% soustředěný a nepodaří se mi dosáhnout seriózních výsledků [2].

Vyplatí se udržovat pořádek na stole a v okolí pracoviště či na ploše počítače. Není potom nutností řešit malicherné problémy typu - „kde jsem včera uložil ten soubor“ nebo „kam na stůl to mám položit, abych se mohl pohnout“ - a mysl zůstane soustředěná.

2.3.5 „Já to udělám“

Máme ve zvyku dělat za ostatní jejich práci, protože si myslíme, že to nedělají dobře nebo proto, že zvládneme udělat všechno sami? Tímto hrdinským smýšlením ztrácíme čas, kterého nemusíme mít nazbyt [6]. Co můžou udělat jiní, ať udělají jiní. Přenechejme vše, co se dá, ostatním. Tzv. delegování je dalším krokem k lepšímu hospodaření s časem.

2.3.6 „Není to perfektní“

Nic není dokonalé. Žijeme v konzumním světě, kde nejdůležitější věcí je byznys. Nikdo dnes nevyrábí perfektní věci, protože to stojí hodně času a peněz a málo kdo si je může dovolit. Není cílem vytvářet dokonalé věci. Cílem je za dané množství času provést co nejlepší výsledek [6].

2.3.7 „Stejně jsem to nezvládl“

Asi hodně lidí zná ten pocit, kdy večer přemílá, kolik toho přes celý den vlastně nezvládli, kolik toho ještě schází [2]. Z takového myšlení vznikají nepříjemné pocity, které velice špatně ovlivňují morálku. Pozitivní myšlení je tedy jednou z dalších věcí, které není radno podceňovat. Mysl je jedna z nejsilnějších zbraní v cestě k produktivitě a efektivitě, je to jeden ze základních stavebních kamenů každé činnosti. Klíčem je využít svou mysl a nenechat ji, aby se obrátila proti nám. Místo toho, aby byla sklenice z poloviny prázdná, ať je z poloviny plná. Když tedy budu večer zase přemýšlet, vzpomenu si raději, kolik mám již hotovo.

2.3.8 „Nechce se mi“

Rozhodně záleží na každém jedinci zvlášť, jak si bude svůj čas nebo život organizovat. Nejsložitější částí je uvedení teorie do praxe, zrušení svých návyků, přizpůsobení se novým. Ten, kdo nechce ničeho dosáhnout nebo pro to nechce nic udělat, tak mu samozřejmě žádná metoda nepomůže.

3 Getting things done

3.1 Potřeba vyšší osobní produktivity

V minulosti byly problémy více viditelné a bylo tak jasnější, co je třeba udělat, kdy to udělat, kdy je práce hotova a kdy je ještě potřeba něco doladit nebo dodělat. Dnes jsou meze jakékoli práce daleko složitěji rozpoznatelné. „Lidé se snaží vyřešit najednou více věcí, ale nemají na to čas, a i kdyby ho měli, nedaří se jim provést je dokonale“ [1, str. 5]. Absence těchto mezí potom způsobují stále větší a větší nárůst práce a povinností. A není to však jenom absence těchto mezí, ale i samostaní lidé si dnes nabírají stále více práce, než lze vůbec stihnout. Tím si způsobují mnoho stresu a jsou znechuceni, což se potom značně projeví na jejich dalším přístupu k těmto činnostem.

Dnešní svět se daleko rychleji mění, než tomu bylo kdysi. Kdysi zaběhnuté návody jak se vyrovnávat s nárůstem informací, byly úspěšné ve své době. Pomocí takového kalendáře si lze osvojit jenom některou část těchto nových a stále se měnících problémů. To samé platí i o zjednodušených prioritách a seznamu zvaném „to-do list“. Zvládat dnes změny úspěšně a přitom zůstat bez problémů vyžaduje nové postupy a nový způsob smýšlení.

„Je zapotřebí efektivní osobní organizační systém, který nevytváří v podvědomí blok k novým projektům a cílům, ale umožní jejich zvládnutí v souladu se zadáním“ [1, str. 9]. Tato výsada se již dnes netýká pouze zkušených businessmanů. Pracovat produktivně, kvalitně, bez starostí a stresu je dnes snažením většiny pracovně zatížených jedinců.

Jedním z těchto efektivních systémů je Getting things done. Tato kapitola čerpá údaje o metodě hlavně z knihy Getting Things Done [1].

3.2 Základní přínos a cíl metody

Getting things done neboli „GTD“ je jedna z nejznámějších metod organizace práce. Autorem metody GTD je David Allen.

Metoda se zaměřuje na zlepšení osobní produktivity, získání více času pro sebe a zjednodušení a zlepšení životního stylu, pomáhá uvědomit si, co právě uživatel vykonává, ale hlavně i to, co nevykonává. Metoda učí uživatele mít nepřetržitou kontrolu nad všemi činnostmi.

Cílem metody je tedy mít plný přehled nad všemi rozpracovanými věcmi a nad vším, co je právě vykonáváno, ne jenom mechanicky a bezmyšlenkovitě zpracovávat jednu příchozí informaci podruhé.

Hlavním přínosem této metody je vytvoření systému, který umožní soustředit se na jednu věc, aniž by vznikla potřeba přemýšlet nad ostatními nedokončenými povinnostmi. Zachytit tak jednoduché i náročné úkoly, které je potřeba udělat dnes, za týden nebo měsíc, a předat je tomuto systému.

Metoda však vyžaduje po uživateli snahu změnit se. Každý má nějaký zlovyk, co se týče organizování práce a tento zlovyk bude třeba změnit, aby bylo možné metodu aplikovat a zvládnout tak jednotlivé techniky.

Většinu principů metody GTD lze využívat nejen v aspektech pracovního, ale i osobního života.

3.3 Autor metody – David Allen

David Allen je nejvýznamnější osobnost v oboru produktivity. Už v dětství se projevil jeho talent v umění mluvit. Do 35 let vystřídal 35 profesí. Pracoval jako číšník, kouzelník, průvodce, učitel karate, distributor vitamínů, kuchař, sklář, prodavač mopedů, manažer benzínky, zahradnické firmy a cestovní kanceláře. Později se stal instruktorem osobního rozvoje. Byl hlavním mluvčím velkých organizací - Světová banka, Ford, New York Life, U.S. Navy, L.L. Bean.

Řízení osobní produktivity začal rozvíjet v roce 1988 při práci na projektu pro zajištění programu pro manažery a vedoucí pracovníky firmy Lockheed.

Založil společnost David Allen & Co. zaměřenou na osobní produktivitu a trénink manažerů, kde je jejím prezidentem. Je jedním ze zakladatelů firmy Actioneer, Inc., zaměřené na výrobu nástrojů pro produktivitu na mobilních telefonech PDA Palm Pilot.

David Allen je autorem tří knih:

- Getting Things Done: The Art of Stress-Free-Productivity (2001),
- Ready For Anything: 52 Productivity Principles (2003), která je souborem článků o metodě GTD, které Allen napsal do bulletinů,
- Making It All Work: Winning at the Game of Work and the Business of Life, volné pokračování knihy první.

3.4 Pět kroků k efektivnímu využití svého pracovního času

Jádrem metody GTD je pět kroků, kterými je potřeba se řídit za každé situace. Jsou to jednoduché zásady, které jsou většinou velmi podobné se základními principy, které již mnozí používají. Tyto principy však GTD výrazně vylepšuje. „Každý proces z těchto pěti kroků je tak kvalitní, jak kvalitní je jeho nejslabší článek. Pokud jeden z těchto článků příliš nekvalitní, pak rozhodnutí učiněné v jakémkoli bodě v závislosti na tomto článku nemusí být správné“ [1, str. 24].

3.4.1 První krok - sesbírání všech aktuálně řešených informací

3.4.1.1 Dostaňte vše ven z hlavy Určitá část mysli neustále přemýšlí nad tím, co je potřeba udělat. Úkoly se do této části sbírají automaticky a ukládají se podobně jako na záznamník. Všechny nevyřešené nebo nezaznamenané úkoly potom zůstávají v paměti. Pokud je paměť alespoň částečně uvolněná, dokáže mozek pracovat soustředěněji a koncentrovaněji. Čím více se však paměť zaplňuje, tím více soustředění a koncentrace klesá, až nakonec mozek není schopen vykonat jedinou operaci. Toto je u většiny lidí hlavní příčinou stresu.

Hlavní a nutnou změnou pro všechny, kteří chtějí metodu GTD aplikovat, je dostat vše ven z hlavy. Každý člověk měl někdy tolik úkolů, že si musel pomoci a vytvořil si jejich seznam. „Většina lidí toto ale dělá až v krajní nouzi, kdy už není vyhnutí“ [1, str. 22]. Metoda GTD radí dělat seznam celý život pro všechny jeho aspekty.

3.4.1.2 Zapište to Každý dnes zvládne věci zapisovat, ale co zapsat a jak zapsat, aby se s tím dalo dále pracovat?

Zapisovat se doporučuje všechno, co lze zhledat neúplným, nekompletním, nebo co by se dalo vlastními silami změnit. Je to úplně vše od miniaturních činností až po ty nejhůře proveditelné. „Neúplné, nekompletní, je vše o čem si myslíme, že bychom to měli udělat. Jakmile si řekneme, že bychom něco měli nebo potřebujeme udělat, stává se to nekompletní“ [1, str. 26].

V této fázi se zapisují všechny položky, které je potřeba udělat, ale ještě nejsou provedeny. Až se rozhodneme, jak se zapsanými položkami naložit a tyto položky zpracujeme, z tohoto seznamu se vyškrtnou.

Je důležité si uvědomit, že všechny úkoly, které se mají vyřešit, se zaznamenávají na vhodném místě, ale nezůstávají v paměti.

3.4.1.3 Kam si úkoly vhodně zaznamenat?

- zápisník, diář
- počítač
- mobilní telefon
- hlasový záznamník
- outlook

3.4.1.4 Uvedení do praxe K disciplíně a sebeovládání žádné externí zařízení nepomůže. Jak tedy zajistit, aby vše fungovalo tak, jak má? GTD radí nosit si všude s sebou svůj záznamník tak, aby bylo možné kdykoli zaznamenat jakékoli myšlenky, které přicházejí na mysl. GTD nedoporučuje více záznamníků pro různé situace, např. jeden používaný pro práci, druhý při relaxaci, třetí při sportování atd., protože se tím zhoršuje zpracovávání zaznamenaných úkolů a orientace mezi záznamníky. Proto je důležité volit záznamník tak, aby svými vlastnostmi vyhovoval všem, nebo co nejvíce činnostem, při nichž bude používán (kancelář, sport, cestování. . .).

3.4.2 Druhý krok - zpracování sesbíraných informací

Nejdůležitějším krokem po zapsání všech úkolů je postupné vyprazdňování záznamníku, vyřazování úkolů. Jinak se záznamník stane pouhým skladištěm neuspořádaných informací.

Vyprázdněním se myslí rozhodnutí, jak s úkolem dál naložit. Neznačená to tedy jeho vyřešení, ale jeho analýza, a pokud se analýzou nevyřeší, začlenění se dále do systému. Nikdy se nevrací zpátky do záznamníku. Tato činnost se provádí tak často, aby bylo možné zobrazit najednou všechny úkoly, které záznamník obsahuje.

Analýzou se myslí rozhodnutí, co udělat s daným problémem. Chybou je organizovat věci bez předchozího rozhodnutí či analýzy a s nekompletními údaji o dané činnosti. Celý model popisuje obrázek 1.

Zpracovávání se řídí dle těchto pravidel.

- Se zpracováváním se začíná vždy od nejmladších položek. Např. u nevyřízené pošty, nezačneme řešit nejdříve tu nejstarší, protože okolnosti se mezitím mohly změnit.
- Zpracováváme vždy pouze jednu položku najednou.
- Položka se nikdy nevrací zpátky do záznamníku.

„Může se stát, že při zpracovávání jednotlivých úkolů si nebudeme vědět rady, odložíme úkol stranou a začneme se zabývat jiným“ [1, str. 122]. Tím ale vznikne hromada odložených úkolů, které se budou stále odkládat na úkor těch jednoduše splnitelných a postupem času se na ně zapomene. Proto je důležité zpracovávat pouze jeden úkol za druhým a úkoly nepřeskakovat.

3.4.2.1 Otázka „Co to je?“ Pokud se nám nebude chtít zjistit všechny podrobnosti nebo nezjistíme, o co se jedná, pak úkol skončí hluboko v našem stole. Přitom je to záležitost, která nezabere mnoho času.

3.4.2.2 Otázka „Vyžaduje to nějakou činnost?“ Jsou jenom dvě odpovědi na tuto otázku.

- „Ne“ pak se budeme dále rozhodovat zda:

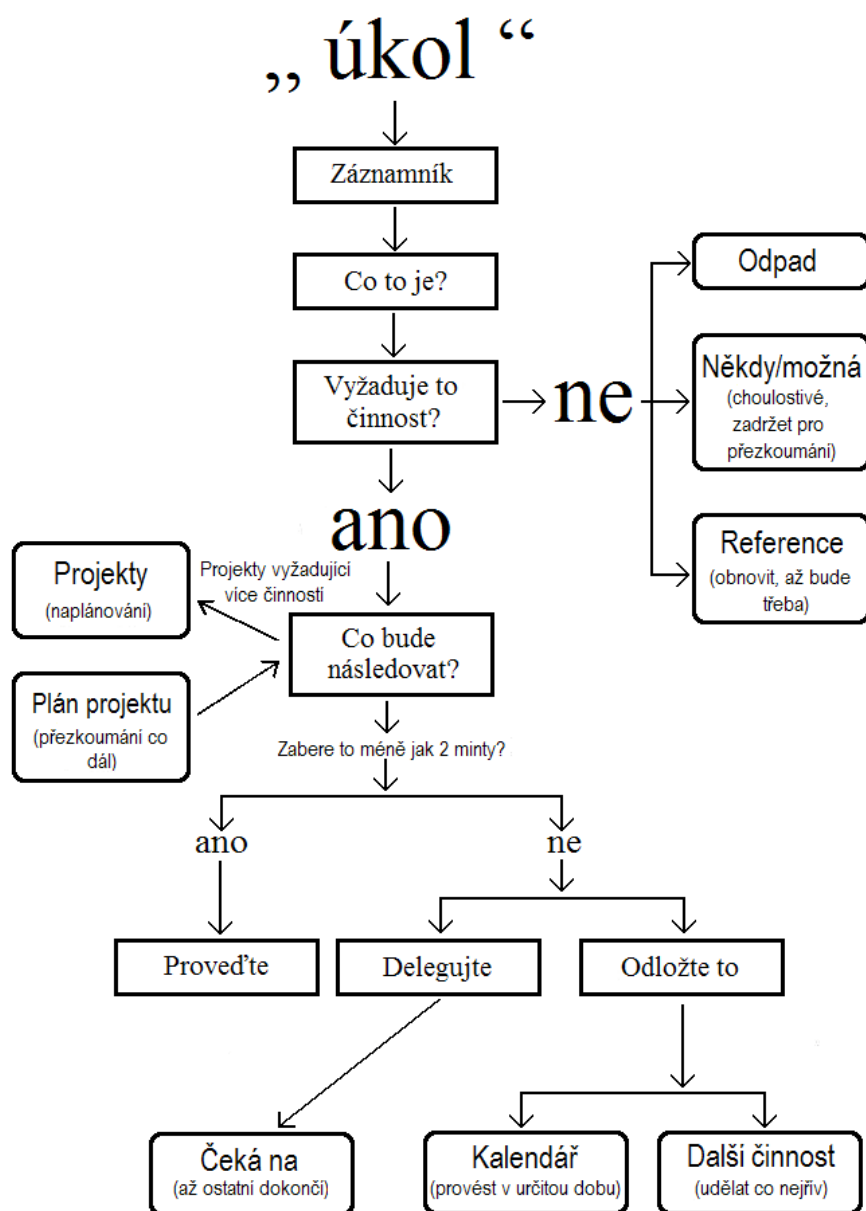
- již tuto položku nepotřebujeme, v tom případě patří do koše,
- v tuto chvíli není vyžadována žádná činnost, ale časem by se mohla objevit, tzv. inkubační stádium,
- tato činnost je informací, která by se nám mohla potencionálně v budoucnu hodit, je to reference.

- „Ano“ v tom případě je pro každou položku třeba určit:

- jaký plán/projekt nebo výsledek jsme si dali za cíl?
- jaká další činnost bude následovat?

Pokud jsme se rozhodli pro větší projekt, vytvoříme si seznam projektů, kde si jej zapíšeme.

Důležité je znát odpověď i na další otázku, a to „Jaký úkol bude následovat?“. „Úkol, který má následovat je činností, která přibližuje vytyčený cíl blíže k realitě“ [1, str. 34].



Obrázek 1: Stromový model druhého a třetího kroku GTD metody [1, str. 32]

Např. chceme prodat své auto, které ale není ve stavu, kdy bychom za něho mohli dostat dobrou cenu. V tomto případě bude následující činnost např. oprava odřené lak, výměna levných opotřebovaných dílů, doplnění kapalin, vyčištění interiéru, umytí karosérie a nakonec podání inzerátu.

Jestliže činnost zabere méně jak 2 minuty, například telefonní hovor nebo odpověď na důležitý email, provede se hned. Pokud ne, je na výběr opět několik možností.

- Jestli usoudíme, že nejsme tím, kdo by se měl touto činností zabývat, úkol předáme, delegujeme.
- Pokud ostatní možnosti selžou, nezbyvá nic jiného než činnost odložit na později, zapsat ji do seznamu mezi další kroky.

3.4.3 Třetí krok - zorganizování činností

Jak ukazuje obrázek 1, z procesu zpracovávání vznikne osm samostatných seznamů, záznamníků.

3.4.3.1 Někdy/možná Sem patří položky, které nevyžadují zrovna žádnou činnost, ale které chceme využít někdy v budoucnu. Je důležité oddělit tyto položky od položek činností vyžadujících, jinak se ve vzniklé směsi nevyznáme a nebudeme vědět, kde začít.

„Může být zajímavé a inspirující udělat si seznam činností, kterých chceme někdy v budoucnu dosáhnout, avšak dnes jsou z nějakého důvodu nerealizovatelné“ [1, str. 32]. Tento seznam je pro nás hnacím motorem, zde se totiž většinou zapisují ty příjemné úkoly, které nás něčím obohacují nebo při kterých relaxujeme. Např. výlety do zahraničí, různá školení, knížky k přečtení atd.

3.4.3.2 Seznam projektů Projekt je úkol, který vyžaduje ke svému provedení více činností. Pokud úkol nejde vyřešit v jediném kroku, stává se projektem. Jestliže toto zůstane opomenuto, jednotlivé kroky k dokončení úkolu setrvávají stále v paměti. Pokud lze i ty nejmenší činnosti rozkouskovat na ještě menší části, musí být rozkouskovány tak, aby všechny úkoly byly nakonec dále nedělitelné. Takové úkoly je potom velice jednoduché uskutečnit a udržovat tak snadno pracovní tempo. Tento seznam slouží hlavně k procházení všech projektů a ujištění se, že ke každému projektu je přiřazen další krok, úkol, který má následovat. GTD doporučuje seznam procházet jednou týdně a připomenout si tak všechny své nevyřešené úkoly, které jsou nyní rozpracované.

To, co se ve skutečnosti vykonává ovšem není projekt. „Lze podniknout pouze kroky, jednotlivé úkoly, které nás pomalu posouvají k cíli a pokud jsou tyto činnosti již hotovy, dostaneme se do fáze, kdy se naše ideály stanou alespoň z části realitou“ [1, str. 38]. Tím se projekt stává hotovým.

3.4.3.3 Seznam plánů projektu Tento seznam slouží pro doplňující informace k jednotlivým projektům v soupisu projektů. Časem vznikne potřeba zapsat si různé detaily o projektech, podrobnosti. Do tohoto katalogu se zapisují:

- činnosti, které bude potřeba vykonat, aby mohl být projekt považován za hotový,
- další krok, který je třeba splnit v cestě za cílem projektu
- aktuální stav projektu,
- a nejdůležitější položkou je cíl, výsledek projektu.

Projekt musí mít přesný a reálný cíl. Pak bude práce na něm zajímavá a efektivnější. Pokud cíl zůstane neurčen, bude velice složité přijít na další krok projektu a nikdy nebude jisté, zda je projekt již úspěšně dokončen. Projekt, který nemá určený cíl se stane nedokončitelným a nepřekonatelným a nakonec skličujícím.

3.4.3.4 Seznam dalších kroků Po zpracování sesbíraných informací se tento seznam společně s kalendářem stane jádrem organizování osobního time managementu. Dalším krokem je jakákoli viditelná, realizovatelná činnost, jejímž splněním se lze dostat k nějakému výsledku.

Většinou soupis nabývá nevídaných rozměrů, takže se vyplatí ho rozdělit do několika kategorií, podle GTD kontextu. GTD kontexty jsou závislé na prostředí a zdrojích, které jsou nutné ke splnění daných úkolů. Například „doma“, „v práci“, „na počítači“, „telefonní hovory“, „nákupy“, „čtení“. Hlavní výhoda rozdělení činností podle kontextu do několika jednotlivých seznamů spočívá v zobrazení jenom těch úkolů, které lze vzhledem k situaci zrovna vykonávat. Pokud jsme zrovna v práci, nezajímají nás úkoly, které máme naplánovány na doma. GTD doporučuje kontexty vytvářet vzhledem k prostředí, podobně jako již výše zmíněné.

Další výhodou rozdělení seznamu podle kontextu je jednoduchost kdykoli a rychle změnit to, na čem se právě pracuje. Pokud se zrovna něco nedaří, nebo z nějakého důvodu nelze pokračovat v právě prováděné činnosti, jednoduše se lze podívat na seznam jiný a pokračovat v práci. Předpokládejme, že hledáme na internetu nějaký typ pracovních informací a v průběhu vyhledávání spojení vypadne. Mezitím, co ostatní využijí tuto událost jako pauzu na kávu, vytáhneme si svůj seznam s kontextem „telefonní hovory“ a malou přestávku strávíme efektivně.

3.4.3.5 Kalendář Sem patří všechny činnosti, které musí být splněny v určitý den a čas. Kalendář je místem, kde lze zaznamenat také informace, které jsou s určitým datem spojeny.

Do kalendáře nenáleží však už nic dalšího. Metoda GTD navrhuje klasický „to-do list“ na celý den. Pracovní celodenní plán není špatnou záležitostí, pokud jej lze kdykoli změnit. „S obrovským množstvím nových záznamů a přicházejících problémů se mění i priority. Proto je celkem nemožné naplánovat si takový seznam dopředu“ [1, str. 40]. Navíc je taková věc ztrátou času a pokud se položky na seznamu odkládají, stává se značně deprimující. Dalším problémem je, pokud budou na celodenním seznamu položky méně důležité, je možné, že v návalu pracovních problémů se pak zapomene na ty opravdu důležité.

3.4.4 Čtvrtý krok - zhodnocení

Jedna věc je úkoly zapsat. Druhá věc ovšem je si na tyto úkoly vzpomenout, když je třeba. Pro mnoho lidí je klíčem osobní organizace práce pravidelné, důsledné procházení a přehodnocování svých úkolů. Takovéto zhodnocování přináší širší přehled nad povinnostmi a zabraňuje, abychom se v projektech ztratili. Seznamem, který se bude velice často procházet, je kalendář. Každý chce být připraven a připomenout si všechny činnosti na další den. „To však neznamená, že úkoly zapsané v kalendáři jsou ze všech hledisek nejdůležitější, jen je třeba je provést“ [1, str. 45].

Je dobrým zvykem si po splnění činnosti z kalendáře zkontrolovat jaké další úkoly zbývají. Po prolistování kalendáře se kontroluje seznam dalších kroků, který obsahuje položky, jež lze vykonat, kdykoli je času nazbyt. Pokud jsou navíc tyto položky seřazeny podle kontextů, procházejí se jenom ty, které se hodí do daného kontextu. Seznam projektů, seznamy „čekám na“ a „někdy/možná“ se doporučuje procházet tak často, aby se o ně nebylo třeba zajímat a nepřidělávaly zbytečné starosti.

3.4.4.1 Pravidelné týdenní zhodnocování „Většina lidí vnímá svou práci pozitivně, když si vyčistí hlavu, dokončí, uzavře a objasní všechny závazky, co si určili nebo které jim určily okolnosti“ [1, str. 47]. Touto fází si mnoho lidí projde jednou za rok. Je dobré provádět pravidelné zhodnocování a objasňování všech informací a činností na seznamech nejméně 1krát týdně. Čím více se bude tento systém metody GTD dodržovat, tím více se stane důvěryhodným. Jedná se hlavně o pocit, že je vše pod kontrolou a přitom hlava zůstává čistá a zbavená všech problémů.

Mezi seznamy, které se doporučuje každý týden procházet patří: „Další kroky“, „Čekám na“ a také „Někdy/možná“.

Týdenní zhodnocování se skládá z:

- Sesbírání a zpracování všech úkolů, problémů a informací,
- prolistování celého systému,
- aktualizování všech seznamů,
- objasnění, uzavření a dokončení problémů.

3.4.5 Pátý krok - realizace

Základem metody GTD je určit podle aktuální situace, který úkol v daném okamžiku udělat. Toto je velice intuitivní činnost, ale s předchozím naplánováním se tato rozhodnutí stávají více jistějšími. „Učinit správná rozhodnutí, kdy kterou činnost provést a přitom si být jistý, že toto je určitě to nejlepší rozhodnutí, by pro nás mělo být výzvou“ [1, str. 130]. V metodě GTD existuje několik technik, pomocí kterých lze tohoto bodu dosáhnout.

3.4.5.1 Čtyři kritéria První technikou je výběr úkolů v závislosti na čtyřech kritériích. Mezi činnostmi se lze rozhodovat:

- Podle kontextu,
- podle času,
- podle energie,
- podle priorit.

Většina činností ke svému provedení vyžaduje nějaké zdroje a určité prostředí. Jenom málo věcí lze udělat kdekoli. Prvním limitujícím faktorem je tedy kontext.

Dalším faktorem je čas. Ne vždy je čas na rozhodování, kterou činnost zrovna udělat, jelikož je zrovna potřeba provést činnost jinou. Pro některé úkoly je také potřeba dostatek energie nebo soustředění. Po velmi náročném dni je mnohem jednodušší výběr.

Pokud předchozí limitující faktory selhaly a činností je pořád na výběr mnoho, přijdou na řadu priority. GTD pro tuto fázi poskytuje další techniku.

- Provést předem stanovené úkoly
- Řešit problémy v pořadí, v jakém se objevují
- Stanovit si, co je třeba udělat

Je dobré se před výběrem jakékoli další činnosti podívat do kalendáře, jestli náhodou není nějaký úkol naplánovaný předem. Pokud existují takové úkoly, je lepší je provádět před ostatními činnostmi.

Jsou problémy, které se objevují nečekaně a musí nebo mohou se pak vyřešit ihned. Mnohdy se každý den objeví nějaké neplánované překážky, které vyžadují alespoň trochu času a energie. Pokud jsou tyto překážky překonávány, je tak automaticky rozhodnuto, že tyto problémy mají větší prioritu než jakékoli jiné úkoly.

Třetí možností je stanovit si úkoly, které je potřeba vykonat. Znamená to provést celý druhý a třetí krok metody GTD. Zpracovat sesbírané informace a zorganizovat je. Při zpracovávání dostanou samozřejmě přednost položky, které lze vykonat do 2 minut. Je to druhá varianta řešení problémů v pořadí, v jakém přicházejí. Jakmile jsou již 2 minutové položky hotové, zůstanou jenom ty složitější, mezi kterými pořád (i vzhledem ke kontextu, času a energii) zůstává možnost výběru. Poslední technikou, která v rozhodování pomáhá je „6-úrovňový model pro přezkoumávání vlastních úkolů“.

3.4.5.2 Šesti-úrovňový model pro přezkoumávání vlastních úkolů Rozhodování mezi úkoly by se mělo v zásadě řídit prioritami. Pro zjištění těchto priorit je však třeba definovat vlastní cíle. GTD používá k analogii tzv. letovou hladinu.

- Vzletová dráha: aktuálně řešené úlohy
- 3.000 metrů: aktuálně řešené projekty
- 6.000 metrů: pole odpovědností
- 9.000 metrů: cíl v úseku jednoho až dvou let

- 12.000 metrů: vidina za období tří až pěti let
- 15.000 a více metrů: životní cíle

Cíle, které se nacházejí vysoko, se skládají z úkolů a činností v nižších letových hladinách. Prioritní jsou tedy ty položky, které se nacházejí nejvýše. Aby se život mohl stát produktivním, je třeba si určit nejvyšší cíle nejdříve. Všechny činy jsou řízeny prioritami z vyšší letové hladiny, proto je třeba začít co nejvýše. Rozhodnout se o způsobu života, životních cílech, a jak tyto cíle vyplnit. Jestliže se nejdříve určí priority z nižších úrovní, může se lehce přihodit, že se tyto cíle postupem doby ukážou jako špatné a všechen čas strávený nad nimi se tak stane promrhaným. Nejdůležitější letovou hladinou je tedy 15.000 metrů.

Vzletová dráha obsahuje všechny úkoly, problémy a činnosti, které se nasbíraly za poslední dobu a v nejbližším termínu musí být dokončeny. Mohou to být většinou ty nejjednodušší činnosti jako schůzky, telefonáty a různá zařizování. K jejich dokončení by podle GTD bylo pravděpodobně potřeba 300 až 500 hodin pokud by se zastavil čas a jiné další úkoly by nepřibývaly.

Ve výšce 3.000 metrů se nacházejí nejbližší projekty, které je relativně jednoduché dokončit a nezaberou mnoho času.

Vzletová hladina 6.000 metrů definuje odpovědnosti, které závisí na zaměstnání, společenském postavení, osobním životě. Mnoho projektů vznikne z povinnosti a odpovědnosti. Mohou to být pracovní záležitosti, ale i záležitosti osobní, např. starost o své zdraví, rodinu, finance, odpočinek atd.

Zatímco předchozí hladiny měly co do činění s aktuálním stavem úkolů, činností, projektů a oblastí zodpovědnosti, od 9.000 metrů a výše jsou hlavní myšlenkou cíle, plány, úmysly a všechny faktory smýšlení o budoucnosti. V 9.000 metrech se může jednat o plány v zaměstnání, v 12.000 metrech o kariéru a osobní úspěchy a v 15.000 metrech o poznání smyslu života a jeho prožití na plno.

3.5 Zhodnocení metody GTD

Jak to již bývá, je celkem nemožné dát 100% úspěšný návod obecně pro každého. Jak tvrdí sám autor této metody, GTD je funkční pouze jako celek a pouze tehdy pokud se řídíme všemi jeho radami. Já si však dovoluji tvrdit, že ne každý využije, nebo mu úplně sedí všechny části a principy GTD. Z osobní zkušenosti vím, že mnoho lidí si některé části této metody přizpůsobuje po svém, nebo nějaké úseky vystřihne úplně. Pro někoho je to příliš časově náročné, příliš detailní, nebo to úplně nesedí pro situaci, v které se jednotlivci zrovna nachází. Na druhou stranu se i autor zmiňuje o tom, že existuje ještě spousta věcí, které sám nezmínil a na které musí každý přijít sám. Nejdůležitější je vyzkoušet si tuto metodu v praxi a pokud se změny, které na GTD provedeme osvědčí, získáme daleko silnější způsob, jak se s touto problematikou vyrovnat, protože bude šitý na míru přímo vám.

Metoda GTD má také pár nevýhod. Pokud se totiž někdo dostane ke všem seznamům, zjistí o dané osobě úplně vše, což může být někdy dosti nepříjemné.

4 Nárvh

4.1 Návrh desktopové aplikace

Návrh desktopové aplikace vychází z velké části z aplikace *Shuffle* pro mobilní telefony s operačním systémem Android. Tato aplikace byla vybrána z důvodu, že poskytovala solidní implementaci metody GTD na této platformě, ale nezajišťovala žádný druh synchronizace. Dle zadání se měla vytvořit aplikace desktopová, mezi jejíž funkce měla patřit i synchronizace s touto mobilní aplikací. Jedná se tedy o jakousi portaci mobilní aplikace na desktop, tudíž i analýza se tomuto značně přizpůsobuje.

4.1.1 O programu Shuffle

Shuffle je aplikace pro mobilní telefony na platformě Android. Základním účelem Shuffle je poskytnout uživateli druh aplikace, která může sloužit jako velmi dobrý pomocník při plánování a organizování osobního času. Inspirací v tomto snažení je metoda GTD. Aplikaci Shuffle navrhnul a vytvořil Andy Bryant. Na vývoji se stále pracuje. Aplikace má své oficiální stránky na internetu [7].

Aplikace se řídí několika základními principy této metody. Hlavní menu je rozděleno do položek:

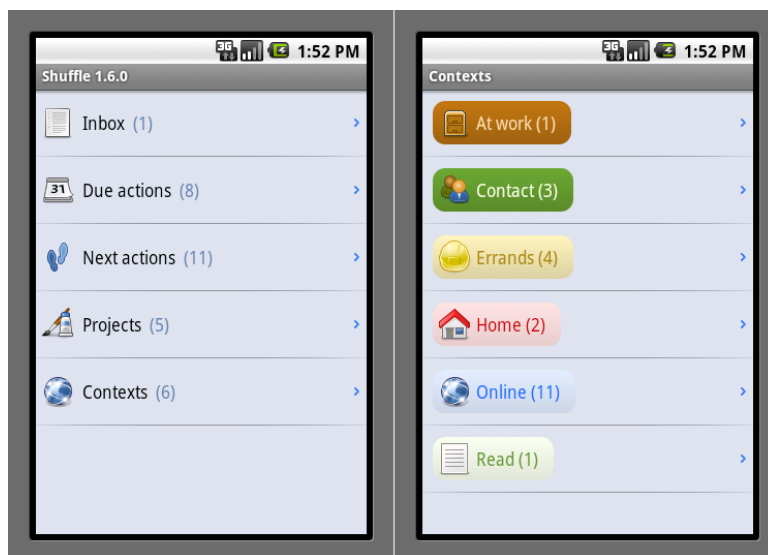
- „Inbox“
- „Due actions“
- „Next actions“
- „Projects“
- „Contexts“

„Inbox“ je klasickým záznamníkem, jak jej popisuje GDT. Zde si lze zapsat všechny příchozí informace, které se během dne nahromadí. Položky v tomto seznamu potom čekají na zpracování a roztřídění. Až se tomu tak stane, lze tento „inbox“ „pročistit“, takže všechny položky, které jsou zpracovány, ze seznamu vymizí.

Menu „due actions“ zobrazuje položky, které jsou naplánované na určitý den nebo hodinu. Shuffle podporuje od vyšších verzí možnost zadat datum a čas počátku a také datum a čas konce. Celé menu je dále rozděleno na tři panely, kde první zobrazuje činnosti, které jsou naplánovány na dnešek, druhý panel činnosti pro následující týden a třetí panel pro měsíc.

„Next actions“ je seznam všech následujících činností. Zde se zobrazují pro každý projekt položky, které v projektu následují.

Záložka „projects“ je opět inspirována prvkem, který definuje metoda GTD. Je zde seznam všech projektů a pro každý projekt lze zobrazit jeho plán. V tomto případě se však plán omezuje pouze na všechny činnosti daného projektu. Činnosti je možno seřazovat podle priorit a určit tak, která položka bude následující akcí v projektu.



Obrázek 2: Ukázka mobilní aplikace

Činnosti mohou být v aplikaci rozděleny i podle GTD kontextů. Všechny kontexty v aplikaci lze zobrazit v menu „contexts“, kde pro každý kontext je možno opět ukázat všechny položky.

Aplikace dále poskytuje různá upozornění a nastavení, která lze propojit s uživatelským rozhraním operačního systému Android. Tyto možnosti však nelze promítnout do desktopové aplikace, proto zůstanou nezmíněny.

Obrázek 2 zobrazuje ukázkou z mobilní aplikace, hlavní menu a seznam kontextů.

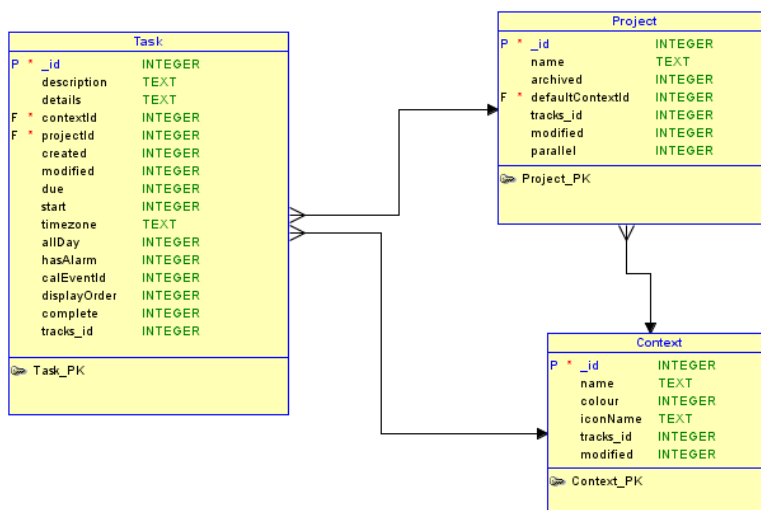
4.1.2 Koncepce

Jak již bylo řečeno výše, několik prvků aplikace Shuffle nevyhovuje mým nárokům na to, jak by taková aplikace měla vypadat. Několik prvků je také nepoužitelných pro desktop. Z těchto důvodů je návrh desktopové aplikace relativně zjednodušen a přizpůsoben více uživatelským požadavkům. Vzhled aplikace by měl samozřejmě zůstat podobný, aby usnadnil uživateli přechod z mobilní aplikace.

Základní vlastnosti budou podobné vlastnostem z aplikace Shuffle. Uživatel bude moci položky vytvářet, zpracovávat, měnit, mazat a označovat za hotové. V menu bude „inbox“, seznam projektů, následujících úkolů a kontextů stejný, jako je tomu v aplikaci mobilní.

Jednou ze změn oproti Shuffle je návrh nové položky v menu, která bude vstupním bodem pro veškeré vytváření nových činností, projektů a kontextů. V závislosti na tom, ubudou pro zjednodušení také možnosti vytvářet tyto položky v jednotlivých hlavních nabídkách. Tato změna velmi zpřehlední celou aplikaci a usnadní orientaci a její naučení.

Další novou nabídkou v menu bude seznam všech kompletních činností. Shuffle toto opět nezobrazuje na jednom místě. Je vhodné ukázat uživateli, čeho již dosáhl, a co má



Obrázek 3: Konceptuální model desktopové aplikace

hotové jednoduše a přehledně. Položky v tomto seznamu bude možno také označit za nekompletní, v případě, že by se uživatel zmýlil a omylem označil položku za hotovou, nebo v případě, že by se nároky pro splnění daného úkolu změnily. Položky z této nabídky půjdou dále pouze mazat.

Zobrazování plánovaných úkolů není v Shuffle řešeno vyloženě špatně, ale postrádá jeden z nejdůležitějších nástrojů jakéhokoli plánování - kalendář. V desktopové aplikaci bude tedy kalendář, který bude graficky odlišovat dny obsahující nějaké činnosti. V kalendáři bude dále možné zobrazit všechny naplánované úkoly pro daný den a bude možné je v této perspektivě měnit. Opět to je přehlednější řešení než v mobilní aplikaci.

Seznam následujících činností bude zobrazovat nadcházející úkoly projektů podle data. Jedná se o další změny, které aplikaci zjednodušují.

4.1.3 Analýza

Jelikož se nejedná o informační systém ani o příliš rozsáhlou aplikaci, přizpůsobuje se tomu i analýza.

4.1.3.1 Datová analýza Datová analýza vychází z velké části z mobilní aplikace. Datová struktura je však o něco zjednodušená, jelikož není potřeba ukládat nastavení a jiné funkce, které mobilní aplikace poskytuje. Aplikace obsahuje pouze tři tabulky - pro ukládání úkolů, projektů a kontextů.

4.1.3.1.1 Konceptuální model Pro ukládání data a času se používá *timestamp*, v databázi uložen jako datový typ *integer*. Konceptuální model ukazuje obrázek 3.

4.1.3.1.2 Lineární zápis tabulek Ačkoli desktopová aplikace nevyužije všechny atributy těchto tabulek, pro správnou funkci synchronizace je nezbytné, aby struktury tabulek byly shodné s mobilní aplikací.

Task (.id, description, details, contextId, projectId, created, modified, due, start, timezone, allDay, hasAlarm, calEventId, displayOrder, complete, tracks_id)

Project (.id, name, archived, defaultContextId, tracks_id, modified, parallel)

Context (.id, name, colour, iconName, tracks_id, modified)

4.1.3.2 Funkční analýza Funkční analýza obsahuje seznam všech funkcí, které se budou v desktopové aplikaci implementovat. Není potřeba dodržovat funkce přesně stejné, jako jsou na mobilní aplikaci, takže se trochu liší. Změna není však natolik rapidní, aby se uživatel, používající aplikaci Shuffle, nedokázal rychle přizpůsobit.

4.1.3.2.1 Seznam procesů

1. Úkoly

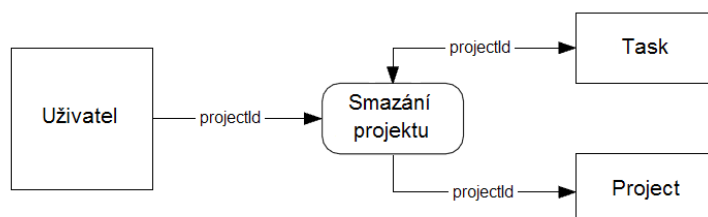
- (a) Vytvořit úkol
- (b) Změnit úkol
- (c) Smazat úkol
- (d) Vyhledat úkoly podle data
- (e) Vyhledat úkoly podle projektu
- (f) Vyhledat nejdřívejší úkoly projektů
- (g) Vyhledat úkoly podle kontextu

2. Projekty

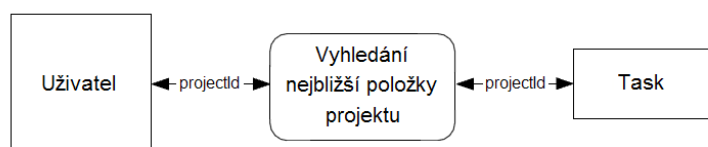
- (a) Vytvořit projekt
- (b) Změnit projekt
- (c) Smazat projekt

3. Kontexty

- (a) Vytvořit kontext
- (b) Změnit kontext
- (c) Smazat kontext
- (d) Vyhledat kontexty podle projektu



Obrázek 4: Smazání projektu



Obrázek 5: Vyhledání nejbližších úkolů projektu

4.1.3.2.2 DFD diagramy Složitější procesy v aplikaci jsou znázorněny pomocí DFD diagramů. Smazáním projektu nebo kontextu se projekt či kontext všech úkolů, které mu náležely, nastaví na výchozí hodnoty. Tento proces zobrazuje diagram 4.

V seznamu dalších akcí se zobrazují nejbližší položky projektů podle data a času. Vyhledávání těchto položek ukazuje diagram 5.

Další procesy v aplikaci jsou více či méně podobné již dříve zmíněným, nebo jsou velmi jednoduché, takže je není potřeba rozebírat dopodrobna.

4.2 Návrh synchronizace

Desktopovou aplikaci bude možné synchronizovat s aplikací mobilní. Existuje několik možností, jak toto provést. Tato synchronizace se však bude řešit tou více komplexnější metodou, a sice použitím serveru.

4.2.1 Koncepce

Server bude hlavním místem, kde se budou data ukládat a uchovávat. Mobilní i desktopová aplikace se k tomuto serveru budou připojovat, poskytovat mu svá data a data ze serveru přijímat. Server si bude uchovávat jednotlivé verze dat a jednotlivé dotazy obou aplikací. Bude řešit konflikty mezi různými verzemi, transakce a paralelní přístup. V případě, že se konflikty na serveru nevyřeší, odešlou se podrobnosti aplikací, kde o výsledku rozhodne uživatel.

Server nebude potřeba spravovat uživatelem. Bude samostanou jednotkou, která bude schopna přijímat požadavky nezávisle na datech a automaticky řešit nastalé problémy.

Desktopová aplikace	Čas	Mobilní aplikace
Změna dat	t_1	-
Synchronizace	t_2	-
200 OK	t_3	-
-	t_4	Synchronizace
-	t_5	200 OK
-	t_6	Projevení změny

Tabulka 1: Synchronizace bez konfliktu

Uživatel se nebude muset starat, co se děje na straně serveru, ale z výstupu aplikace pozná, jak se daná situace na serveru vyřešila.

Dotazy na server budou anonymní. Jakákoli aplikace se bude moci k serveru připojit a synchronizovat se s jeho daty, případně rozhodnout konflikty mezi verzemi, pokud nastanou. Aplikace tedy nebudou muset řešit přihlašovací údaje, registraci, autorizaci atp.

V aplikaci bude možné zadat různé adresy serverů a synchronizovat se s jakýmkoli jiným serverem.

Požadavkům pro synchronizaci nejlépe vyhovuje systém pro správu verzí a bude tedy použit.

4.2.2 Analýza

4.2.2.1 Řešení konfliktů Pro zjednodušení předpokládejme, že jde o konflikt mezi dvěma aplikacemi, tedy mobilní a desktopovou. Obě aplikace jsou nyní ve stavu, kdy jsou synchronizovány a jejich data jsou totožná. V tomto případě mohou nastat tři situace:

- Desktopová aplikace provede změny a synchronizuje se se serverem. Mobilní aplikace se synchronizuje se serverem. Změny provedené desktopovou aplikací se projeví v aplikaci mobilní. Vše proběhne bez problému, jak ukazuje tabulka 1.
- Druhá možnost je opakem první. Mobilní aplikace provede změny a synchronizuje se. Hned poté se i desktopová aplikace synchronizuje a její data se změní do stavu, ve kterém jsou data aplikace mobilní. Opět bez konfliktu. Totožné s tabulkou 1.
- Mobilní a desktopová aplikace provedou změny najednou. Mobilní aplikace provede synchronizaci, která se provede bez problémů. Nyní chce i desktopová aplikace synchronizovat data, ale její pokus skončí konfliktem. Tuto situaci nastiňuje tabulka 2.

Z tohoto příkladu lze vidět, že operace synchronizace je atomická, nerozdělitelná jednotka, je to transakce. V prvních dvou situacích server rozpozná, že příchozí data aplikace mají verzi vyšší, než data, které má uložena. Uložená data přepíše novějšími a verzi zvýší. Starší verze však pořád uchovává. Při dotazu druhé aplikace mají příchozí data naopak zase verzi nižší, než data na serveru, takže server jenom přepoše verzi vyšší.

Desktopová aplikace	Čas	Mobilní aplikace
Změna dat	t_1	Změna dat
Synchronizace	t_2	-
200 OK	t_3	-
-	t_4	Synchronizace
-	t_5	Konflikt

Tabulka 2: Konfliktní synchronizace

V poslední situaci při prvním dotazu server verzi zvýší. Při druhém dotazu však přicházející data mají stejnou verzi jako data na serveru, ale přesto se liší. Nastává konflikt. Server přepošle zprávu o konfliktu aplikaci. Aplikace zobrazí uživateli rozdíly mezi daty:

- Položky, které byly vytvořeny na mobilní aplikaci, ale které v desktopové aplikaci nejsou (mohly být i smazány)
- Položky, které byly vytvořeny v desktopové aplikaci a mobilní aplikace je neobsahuje
- Položky, které jsou na mobilní i na desktopové aplikaci, ale liší se jejich vlastnosti (byly změněny uživatelem na jedné z aplikací)

Uživatel o každé položce rozhodne a výsledek se přepošle serveru, který verzi zvýší. Když se potom druhá aplikace snaží synchronizovat bez toho, aby předtím provedla nějaké změny, získá bez problémů data, která vznikly vyřešením konfliktu. Data obou aplikací jsou v tomto bodě opět totožná a celý cyklus potom pokračuje zase od začátku.

5 Implementace

5.1 Implementace desktopové aplikace

Pro implementaci desktopové aplikace jsem zvolil skriptovací jazyk *Python*. Pro tento jazyk existuje několik knihoven na tvorbu GUI. Mezi nejznámější patří: *GTK*, *Qt*, *Tk*, *OpenGL*, *WxWidgets*. Tyto knihovny jsou většinou napsané v jiném jazyce a existuje pro ně přestavba pro jazyk *Python*. GUI této aplikace je tvořeno pomocí *PyQt*.

5.1.1 PyQt

PyQt kombinuje výhody jazyka *Python* a úspěšné knihovny *Qt*. Tato knihovna je jedna z nejmocnějších knihoven pro tvorbu grafického rozhraní vůbec. Většinou existují dva druhy grafických knihoven. Ty, které neobsahují mnoho předdefinovaných tzv. „widgetů“, ale poskytují velké možnosti v nastavení a tvorbě GUI, a ty, které jsou přesným opakem. *Qt* skloubilo výhody obou. Poskytuje mnoho grafických prvků a dává vývojáři velký prostor a mnoho možností a nastavení při tvorbě GUI.

PyQt bylo vyvíjeno firmou Riverbank Computing, která se však před nedávnem nedohodla s autorem knihovny *Qt* Nokií. Je implementováno jako sada modulů pro *Python*. Obsahuje přes 300 tříd a 6000 metod. Ve verzi 3 bylo *PyQt* dostupné pouze pro operační systém *Unix*. S verzí 4 se *PyQt* stalo multiplatformní. Kromě toho byl od této verze zaveden „model/view framework“, který se nechal inspirovat u webových aplikací.

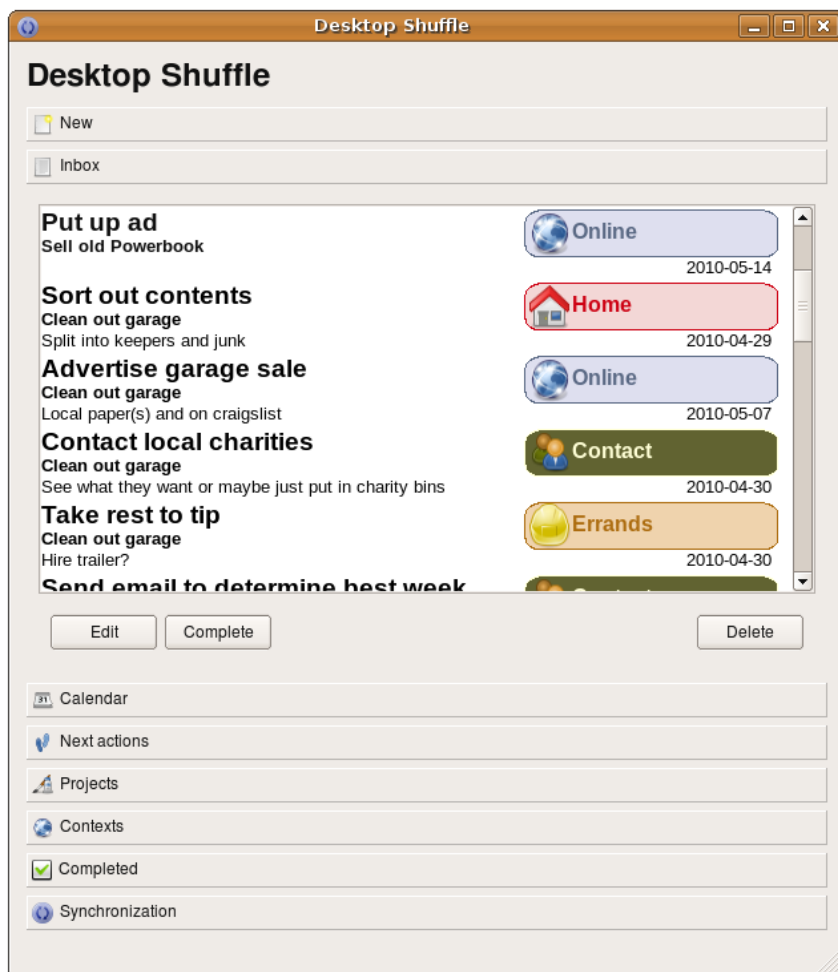
PyQt je kvůli své velikosti rozděleno do několika modulů: *QtCore*, *QtGui*, *QtNetwork*, *QtXml*, *QtSvg*, *QtOpenGL*, *QtSql*. Dokumentaci k *PyQt* lze nalézt na stránkách Riverbank Computing [8].

5.1.2 SQLite

SQLite je relační databázový systém, který se vyznačuje poměrně malou velikostí a svou jednoduchostí. Na rozdíl od většiny jiných SŘBD, pro které je potřeba na serveru vytvořit samostatný proces, který běží po celou dobu, dokud není ukončen, pracuje *SQLite* zcela jinak. Je to pouze malá knihovna, na kterou se aplikace odkazují, tudíž nevytváří zcela nový proces, ale je používána jenom pokud je třeba v procesu dané aplikace. Všechna uživatelská data *SQLite* ukládá do jediného souboru.

Kvůli své velikosti má tato knihovna také jisté nevýhody, a proto není vhodná pro použití ve velkých aplikacích nebo informačních systémech. Není to plnohodnotný SŘBD a nemá některé základní funkce a základní datové typy. Asi nejcitelnější může být absence cizího klíče. Pro menší aplikace se však hojně používá.

Desktopová aplikace využívá *SQLite*. Jedním z důvodů je jednodušší návrh a implementace synchronizace. Jak již bylo řečeno, data jsou uskladněna v jediném souboru na jediném místě, což velmi usnadňuje možnost synchronizace mezi dvěma takovými úložišti. Operační systém *Android* využívá také *SQLite*, takže data uložená v mobilní aplikaci půjde jednodušeji synchronizovat, než by tomu bylo v případě využití velkého SŘBD.



Obrázek 6: Ukázka desktopové aplikace

Dalším důvodem je, že desktopová aplikace je poměrně malá a funkce větších databázových systémů jsou tedy zbytečné a zpomalovaly by chod aplikace. Z výše uvedeného návrhu se má jednat o jednoduchou a rychlou aplikaci a využití SQLite do tohoto výborně zapadá.

5.1.3 Realizace

Desktopová aplikace implementuje všechny funkce z návrhu. Grafické rozhraní bylo vytvořeno co nejjednodušeji a uživatelsky co nejpříznivěji. Nejlépe to zobrazuje obrázek 6. Je zde využito Qt delegátů pro vytváření vlastních grafických prvků. Kvůli snadnější orientaci uživatele jsou obrázky ponechány stejné jako jsou na mobilní aplikaci.

Pro instalaci této aplikace jsou potřeba balíčky: *PyQt*, *sqlite3*, *Git 1.7* a *GitPython*.

5.2 Implementace synchronizace na desktopové aplikaci

V návrhu synchronizace se rozhodlo, že nejlépe vyhovuje požadavkům systém pro správu verzí. V současné době si získává stále větší popularitu verzovací systém Git. Rozhodl jsem se do své aplikace tento systém použít.

5.2.1 Verzovací systém Git

Git je decentralizovaný systém pro správu verzí, jehož autorem je Linus Torvalds, který Git vyvinul původně pro vývoj Linuxového jádra. Proto je Git uzpůsobený zejména pro vývoj OSS. Neurčuje žádný hlavní společný server, žádné role ani odkud je možno začlenit své změny mezi ostatní. Je oblíbený především kvůli tomu, že uživatelé oproti ostatním systémům pro správu verzí poskytuje dostatečnou svobodu. Informace a návody na tento verzovací systém lze nalézt na oficiální wiki [9].

5.2.2 Realizace

Jelikož je Git distribuovaný systém, je tedy decentralizovaný, celkem dobře zapadá do celého systému. Ve skutečnosti může být mobilních nebo desktopových aplikací více, ale pro zjednodušení budu dále předpokládat vždy jenom jednu aplikaci od každého. Jsou tedy dvě úložiště dat, nezávislé na sobě, na kterých se lokálně provádějí změny, a čas od času vznikne potřeba tyto dvě databáze sesynchronizovat. Z návrhu je jasné, že je potřeba vytvořit ještě další, třetí úložiště, které se bude nacházet na serveru, aby se mohly obě aplikace kdykoli jednoduše připojit a synchronizovat. Vznikly tedy ideální podmínky k využití distribuovaného verzovacího systému Git.

Desktopová aplikace používá tedy tzv. „Git repository“. Aby mohl být Git používán v Pythonu, využil jsem sadu obalovacích tříd *GitPython*. *GitPython* umožňuje jednodušší používání funkcí verzovacího systému v jazyce Python.

Aplikace postupuje stejným způsobem, jakým by postupoval uživatel při obvyklém používání Gitu. Nejdříve si vytvoří nový Git repositář, pokud ještě neexistuje. K repositáři přiřadí všechny soubory, které budou podstupovat synchronizaci a všechny operace potvrdí commitem. Ke commitu lze přiřadit i jednoduchou zprávu o změnách, které byly provedeny. Tato zpráva by však neměla být delší než dvě věty. Na základě toho se doporučuje commity provádět tak často, aby se provedené změny daly do dvou až tří vět shrnout. V adresáři, kde se repositář vytváří, se vytvoří nová složka, která o repositáři uchovává všechny informace. Pokud je tato složka smazána, zaniká tím i repositář.

```
git init
git add shuffle.db
git commit -m 'vytvoreni_repositare'
```

Výpis 1: Vytvoření nového repositáře

S vytvořením commitu, se verze v repositáři zvyšuje. Každý commit má svůj unikátní identifikátor tzv. *UUID*. Lze se tedy na něj kdykoli odkázat a zobrazit starší verze a provedené změny. Před každým commitem je potřeba přidat všechny soubory, jejichž změnu má repositář zaregistrovat.

V závislosti na uživatelských požadavcích aplikace vytváří vzdálené repositáře, vůči kterým je později synchronizace prováděna. Git si tento vzdálený repositář ukládá do samostatných větví.

Při vytváření jakékoli aplikace časem vznikne potřeba vyvíjet od sebe více nezávislých verzí. K tomuto účelu používá Git právě tyto větve. Celý repositář si lze pak představit jako strom, kde větve jsou nezávisle na sobě vyvíjené verze. Každá větev potom obsahuje uzly, což jsou jednotlivé commity, a list, což je poslední provedený commit. Výchozí větví každého repositáře je větev *master*.

```
git remote add server git: //server/repository
git fetch server
```

Výpis 2: Vytvoření vzdáleného repositáře

Při vytvoření vzdáleného repositáře, v tomto případě pod názvem *server*, vznikne tedy nová větev. Zároveň jsou data z tohoto repositáře stažena a uložena do této větve (příkaz *fetch*).

Uživatel chce nyní synchronizovat se zadaným repositářem. Git tedy porovná verze a na základě toho se pokusí data sloučit sám. Vzdálený repositář však může obsahovat také několik větví, proto je potřeba určit s jakou větví se má Git pokusit data sloučit.

```
git merge server/master
```

Výpis 3: Pokus o synchronizaci se vzdáleným repositářem

Jelikož je *server* implementován, aby obsahoval pouze jedinou větev, aplikace zadá větev výchozí. Pokud se sloučení podaří, v aplikaci se projeví výsledky a uživateli je zobrazena zpráva o úspěchu.

Pokud se sloučení nepovede, Git s tím tedy sám nic nezmůže a je na uživateli, aby rozhodl o výsledku. Aplikace zkontroluje, jaké data obsahuje vzdálený repositář, a porovná je se svými daty. Uživateli zobrazí rozdíly mezi oběma repositáři. Přejít mezi větvemi se provádí příkazem *checkout*.

```
git checkout server/master shuffle.db
```

Výpis 4: Zjištění obsahu vzdálené databáze

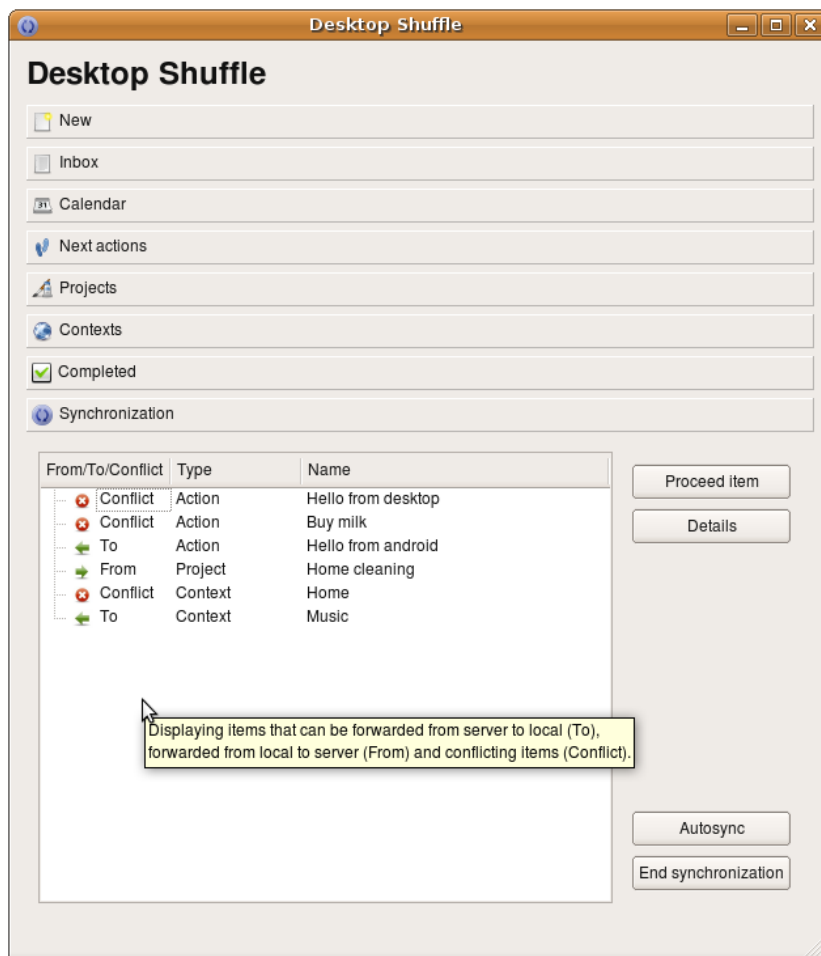
V tomto případě se však repositář nepřesune na větev *server*, ale pouze nahraje stav souboru *shuffle.db*, takže aplikace může porovnat rozdíly.

Nakonec uživatel vyřeší konflikt a aplikace provede commit. Současně se i o výsledku dá vědět vzdálenému repositáři na serveru. To platí i v případě, kdy ke konfliktu nedojde.

```
git add shuffle.db
git commit -m 'konflikt_vyresen'
git push server
```

Výpis 5: Vyřešení konfliktu

Ukázka synchronizace na desktopové aplikaci, kdy dojde ke konfliktní situaci, je zobrazena na obrázku 7.



Obrázek 7: Ukázka konfliktu na desktopové aplikaci

5.2.3 Implementace serveru

Jednou z možností bylo využít již existujícího serveru¹. Požadavkům v návrhu však lépe vyhovuje vytvoření serveru vlastního. Až na několik výjimek se vytváření veřejného repozitáře na serveru podobá vytváření běžného repozitáře.

Implementace využívá serveru *apache2* a balíčku *gitweb*. Gitweb je webové rozhraní pro Git napsané v jazyce Perl a je nezbytné pro procházení repozitáře na serveru.

Po vytvoření se lze na vzdálený repozitář na serveru jednoduše odkázat. Git k tomu používá svůj vlastní protokol.

Více informací o implementaci Git repozitáře na serveru lze nalézt na internetu. [10]

¹<http://github.com/>

5.3 Implementace synchronizace na mobilní aplikaci

Mobilní aplikace je implementována na platformě Android. Jelikož pro operační systém Android neexistuje plnohodnotná implementace Gitu, je synchronizace řešena trochu jiným způsobem, než na desktopové aplikaci.

5.3.1 Android

Android je operační systém vyvinutý společností Android Inc., kterou později koupila firma Google. Je založený na Linuxovém jádře, které docela razantně modifikuje pro své potřeby. Na toto téma vzniklo mnoho diskuzí spekulujících, zda lze Android ještě považovat vůbec za Linux.

Vývoj aplikací na této platformě zajišťuje sada knihoven napsaných v jazyce Java, takže je lze vyvíjet hlavně v tomto jazyce. Hlavním nástrojem pro vývoj je Android SDK, které poskytuje emulátor mobilního zařízení. Existuje také ADT plugin pro IDE Eclipse, který celý vývoj výrazně zjednodušuje. Jak již bylo řečeno výše, aplikace na Androidu používají pro ukládání dat databázi SQLite. Další informace o vývoji aplikací na této platformě lze nalézt na oficiální stránce [11].

5.3.2 Realizace

Implementace synchronizace na mobilní aplikaci používá vzdálené mapování příkazů verzovacího systému Git. To znamená, že aplikace komunikuje se serverem tak, že po zadání dotazu na určité URL se na serveru vykonají požadované operace. Ke komunikaci používá mobilní aplikace knihovny *Apache Http Client*, které se k tomuto účelu v aplikacích pro Android často využívají. Knihovny umožňují jednodušší vytvoření dotazů na server a získání jeho odpovědi.

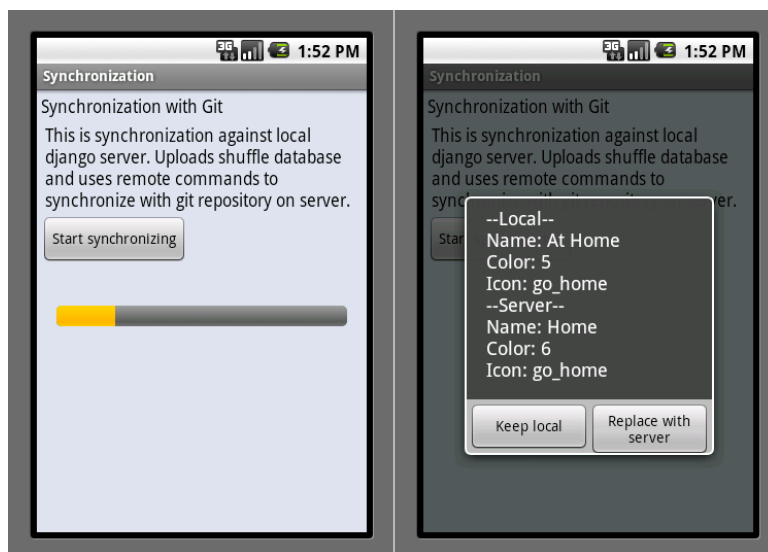
Pro vykonávání příkazů na serveru je použit webový framework napsaný v jazyce Python, Django. Tento framework je velice efektivní a rychlý a pro toto využití dostatečně postačuje.

Synchronizace ze strany mobilní aplikace vytváří ve vzdáleném repositáři na serveru novou větev, která je pak slučována s hlavní větví. Vytvořením nové větve se vytvoří větev, která je identickou kopií výchozí větve.

Celý proces začíná žádostí uživatele synchronizovat data. Mobilní aplikace odešle požadavek a k dotazu připojí databázový soubor, který obsahuje uživatelem provedené změny. Na serveru se repositář přesune do nové větve, kde je připojený soubor nahrán a celá operace je potvrzena commitem.

```
git checkout mobilni_aplikace
git add shuffle.db
git commit -m 'změny_mobilní_aplikace'
```

Výpis 6: Nahrání změn do nové větve vzdáleného repositáře



Obrázek 8: Ukázka synchronizace na mobilní aplikaci

Ve stejném požadavku se server pokusí obě větve pomocí Gitu sloučit. O výsledku operace je mobilní aplikace informována. Pokud je výsledkem úspěch, vyžádá si aplikace sloučený soubor a provedené změny projeví. Není potřeba provádět commit pokud operace sloučení skončí úspěšně, Git toto provádí automaticky.

```
git checkout master
git merge mobilni.aplikace
```

Výpis 7: Pokus o sloučení hlavní větve s vedlejší

Jestliže se sloučení nepodaří, mobilní aplikace odešle dotaz na porovnání obou verzí databázových souborů. Server zjistí rozdíly a výsledek odešle aplikaci ve formě XML, která ho zpracuje a zobrazí dané konflikty uživateli.

```
git checkout mobilni.aplikace shuffle.db
```

Výpis 8: Zjištění rozdílů mezi databázemi

Jakmile je konflikt rozhodnut, je na server zpět odesláno opět XML, které obsahuje výsledné uživatelské rozhodnutí. Server dané změny provede, potvrdí commitem a mobilní aplikaci dá vědět, že činnost skončila úspěšně. Mobilní aplikace si potom zažádá o výsledný soubor a zobrazí konečné změny. Při provádění změn se repositář nacházel pořád ve větvi hlavní, není tudíž potřeba slučovat tuto větev s větví, kterou používá mobilní aplikace.

Jak ukazuje obrázek 8, synchronizace je řešena jako asynchronní proces, kde ve výsledku se konflikt uživateli zobrazuje pomocí dialogů.

6 Závěr

Tato práce měla za úkol navrhnout a implementovat GTD aplikaci s možností synchronizace na mobilní telefon. Výsledná realizace splňuje všechny tyto požadavky jednoduše a srozumitelně. Aplikaci lze bez rozpaků používat pro metodiku GTD. Synchronizace funguje bez problémů a celé řešení je podáno elegantně. Jako spoustu věcí, šlo by samozřejmě i tuto aplikaci vylepšit. Aplikace by mohla obsahovat hezčí, barevnější a složitější grafické prvky nebo nové funkce. Pro účel bakalářské práce však toto řešení postačuje.

Práci hodnotím pro mě jako celkově zdařilou. Byla mi přínosem hned v několika hlediších. Objevil jsem mnoho nových technologií a naučil se řešit mnoho nových problémů. Poznal radost z toho vytvořit něco vlastního, fungujícího a užitečného. V této bakalářské práci jsem se o tyto znalosti a zkušenosti podělil.

Jelikož jsem před touto prací problematiku osobní produktivity a organizace práce a času znal pouze okrajově, čím více jsem do ní pronikal, tím více mě zaujala. Třešničkou na pomyslném dortu byla pak metoda Getting Things Done, kterou bych po napsání této práce chtěl aplikovat do praxe. Nesčetněkrát jsem si ve svém životě vytyčoval cíle, kterých bych chtěl dosáhnout. Tyto cíle však byly odloženy stranou v následku urgentních věcí, nebo byla jejich realizace na tolik nereálná, že byly nakonec zapomenuty. Při studování metody GTD jsem se několikrát pousmál tomu, že nejsem sám, kdo má podobné problémy, a zaradoval se, že již existuje návod na to, jak tyto problémy vyřešit a jak je překonat. Několik lidí v mém okruhu již metodu praktikuje a pár z nich zaznamelo pokles účinnosti této metody s úbytkem práce a úkolů. Ja si však myslím, že tato metoda neřeší pouze pracovní aspekty života, ale lze ji aplikovat i do života osobního. Pokud je tedy nedostatek pracovních úkolů, je o důvod víc k radosti, protože se lze věnovat těm příjemnějším úkolům a činnostem, které udávají smysl našeho života.

7 Reference

- [1] ALLEN, David. *Getting Things Done : The Art of Stress-Free Productivity* USA : Penguin Group, 2001. 288 s. ISBN 01420.00280.
- [2] MÁRA, Pert. *Petr Mára: Mac OS X, GTD, Web 2.0, Lifehacks, iPhone* [online]. 12.9.2007 [cit. 2010-05-05]. Jak žít úspěšný život a jak se motivovat v 10 krocích! . Dostupné z WWW: <http://www.petrmara.com/blogger/2007/09/p1050105jpg.html>
- [3] MÁRA, Petr. *Petr Mára : Mac OS X, GTD, Web 2.0, Lifehacks, iPhone* [online]. 2010 [cit. 2010-05-03]. Dostupné z WWW: <http://www.petrmara.com/blogger/>
- [4] Jan Melvil Publishing s.r.o. *Mít vše hotovo : Getting Things Done* [online]. 2010 [cit. 2010-05-04]. Dostupné z WWW: <http://www.mitvsehotovo.cz/>
- [5] *Wikipedia : The Free Encyclopedia* [online]. 2009, 1.5.2010 [cit. 2010-05-05]. Time management. Dostupné z WWW: http://en.wikipedia.org/wiki/Time_management
- [6] CONSTANDSE, Rodger. *Time Thoughts : Resources for Personal and Career Success* [online]. 2004, 2007 [cit. 2010-05-06]. Time management. Dostupné z WWW: <http://www.timethoughts.com/time-management.htm>
- [7] BRYANT, Andy. *android-shuffle : A GTD personal organizer for the Android platform* [online]. 2010 [cit. 2010-05-03]. Dostupné z WWW: <http://code.google.com/p/android-shuffle/>
- [8] Riverbank Computing Limited . *Riverbank : Computing Limited* [online]. 2010 [cit. 2010-05-04]. Dostupné z WWW: <http://www.riverbankcomputing.co.uk/>
- [9] *Git Wiki* [online]. 2010, 30 April 2010, at 10:55 [cit. 2010-05-04]. Dostupné z WWW: https://git.wiki.kernel.org/index.php/Main_Page
- [10] *HowtoForge : Linux tutorials* [online]. 2009 [cit. 2010-05-03]. How to install a public git repository on a debian server. Dostupné z WWW: <http://www.howtoforge.com/how-to-install-a-public-git-repository-on-a-debian-server>
- [11] Google Inc. *Android Developers* [online]. 2010 [cit. 2010-05-04]. Dostupné z WWW: <http://developer.android.com/>